November/December 1996 Vol. 4 Number 3 $5.00US (North America)

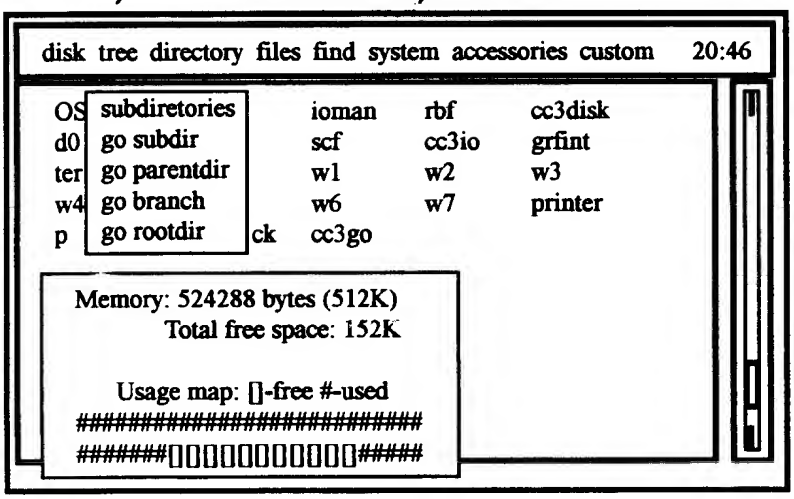
the world of 68' micros

Support for Motorola based computer systems and microcontrollers, and the OS-9 operating system

This issue is dedicated to the memory of Ed Gresick, owner of Delmar Company, who passed away in October.

C. Dekker's CoCoTop...

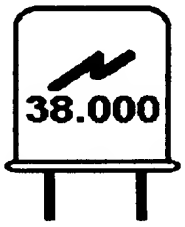
An easy alternative to Tandy's Multi-Vue for OS-9!



CONTENTS

From the editor	2
From our readers	3
Floppies Revisited — Frank Swygert	4
The Need for Speed — A. Hsiao	5
Microware Notes — Boisy Pitre	8
More Letters	8
The Zen of CoCo Programming — Chet Simpson	10
Operating System Nine — Alan Dekok	12
Modifying Process Descriptors — Mark Heilpern	14
G-Windows 3.0 Press Release — GESPAC	15
CoCoTop Review — Frank Swygert	16
All About CoCo ROM Packs — Marty Goodman	18
Advertsers Index	17

THE NEED FOR SPEED...



Soup up your CoCo with a crystal change... run 33% faster, even in DECB!

POSTMASTER:
If undeliverable return to:
FARNA Systems PB
Box 321
WR, GA 31099

If your address is incorrect, send me a postcard!

the world of 68' micros

Publisher:
FARNA Systems PB
P.O. Box 321
Warner Robins, GA 31099-0321

Editor:
Francis (Frank) G. Swygert

Subscriptions:
US/Mexico: $24 per year
Canada: $30 per year
Overseas: $50 per year (airmail)
Back and single issues are cover price. Canada add $0.50 per issue ($1.00 max) additional shipping, overseas add $2.00 per issue. Bulk/ newsstand orders available.

microdisk:
Companion to magazine. Contains program listings, shareware, and freeware as mentioned in magazine text. Three issues per year.
US/Mexico: $20 per year ($8 single)
Canada: $25 per year ($10 single)
Overseas: $40 per year ($15 single)

Advertising Rates:
Contact publisher. We have scales to suit every type of business. Special rates for entrepreneurs and "cottage" businesses.

Contributions:
Any and all contributions welcome. Submission constitutes warranty on part of the author that the work is original unless otherwise specified. Publisher reserves the right to edit or reject material without explanation. Editing will be limited to corrections and fitting available space. Authors retain copyright. Submission gives publisher first publication rights and right to reprint in any form with credit given author.

General Information:
Current publication frequency is bi-monthly. Frequency and prices subject to change without notice. All opinions expressed herein are those of the individual authors, not necessarily of the publisher. No warranty as to the suitability or operation of any software or hardware modifications is given nor implied under any circumstances. Use of any information in this publication is entirely at the discretion and responsibility of the reader.

All trademarks/names property of their respective owners

The publisher is available via e-mail at: dsrtfox@delphi.com

A message from the editor...

What a hectic time it has been recently! My military base just went through an **Operational Readiness Inspection**. For those of you who don't know, this is a full test of a base's abbility to operate under wartime conditions. Every effort is made to simulate actual wartime events. The base functions are then "put through their paces", acting just as if there was a war going on.

For me, being a Civil Engineer member, it means going out in the field for a week and setting up, maintaining, and defending a "tent city". We go out and set up a camp, just as we would if called out to duty. In reality, we would stay in this camp as well as house pilots, ground crew, and everyone else needed to support the flying operations and personnel. We train for a worse case scenario, which is to go out to a remote airstrip with few or no support buildings. Just for the record, our alternate job is to be additional manpower for an established base that is supporting attack operations.

What all this means is I've been out working 12 hour shifts for a week doing some rather exhausting work. As I write, that was two weeks ago, and I finally feel fully recovered! Uou'll be happy to know all the work didn't go unnoticed, as Robins Air Force Base Civil Engineer Group (78CEG)was ranked "Excellent" (ranking, from best to worst, are: Outstanding, Excellent, Satisfactory, and Unsatisfactory).

Luckily, I anticipated the excercise and started work on the magazine early, with just a few items (such as this column) left to finish it up. No late issue this time!

We didn't have a **CoCoFest in Atlanta** this time. Due to the low attendance last year, the Olympics, and amount of effort by a few volunteers to put on a full fledged fest, it just wasn't feasible. The Atlanta Computer Society did, however, host a picnic. The picnic turned out okay, with a few people coming from outside the state (congratulations to Carl Sefik.. he drove down from Michican!). I can't help bu think it would have turned out MUCH better, with a higher attendance, had the weather not taken a sudden turn that Saturday morning. The chill and especially the wind took everyone by surprise! My family and I attneded early. Many others waited until the afternoon, after we had left (we left around 1:30 to run some other errands whil in Atlanta). Many people decided to wait the wind and chill out, showing up between two and four for a few hours of fellowship with other diehard CoCo enthusiasts. Carl England and family pulled up just as we were leaving!

Next year, there will at least be another picnic, though grander plans may be in the works. I am gong to talk to ACS about hosting an **"Eight Bit Bash"**, with clubs and vendors supporting all true eight bit computers and direct descendants being invited. This doesn't mean the 16 bit OS-9 machines will be excluded, just not emphasized. Maybe some older 16 bit Macs and Atari's will be on hand also. Mind you now, this is just an idea... reading it here may be the first ACS hears about it! I'll keep readers posted should such an event become a reality.

I'll close this issue with holiday well wishes for all. I am Christian and celebrate the applicable holidays, but I hold no ill will toward anyone who believes in a single devine being, no matter what there religion. For all I know, *He* has a reason for the diverse methods of worship which we are not privy to. ***So no matter what your religious beliefs, I hope you find peace and joy through this holiday season.***

Messages from our readers...

In the last issue you printed a letter about my printer problems. I used a Blue Streak Ultima serial/parallel adapter (can't find anymore, anybody out there have some they want to sell?) - with a Brother HL-630 laser printer (which emulates an Epson FX850. The Ultima does not pass line feeds so I use PRINT #-2, CHR$(27); CHR$(10) to line feed at the end of each complete print line. Somewhere in the front of the program I use PRINT #-2, CHR$(27); CHR$(48) to set the line spacing to 1/8 inch. This works very well with the CoCo. Needless to say I am happy now and can print W-2 forms and payroll/vendor checks on the laser with my CoCos.

Ken Greiner
1147 Columbus Pike #209
Delaware, OH 43015

Ken, the Brother is an excellent printer and one of the more affordable. I like both the straight paper path AND the Epson emulation. As for serial/parallel converters, many general computer supply houses have them. They cost a bit more the the Blue Streaks (which originally sold for around $30), but will work with any computer. They require a CoCo 4 pin to DB-25 cable. Dalco electronics sells them for $44.50 (#38725, 1-800-445-5342). They also have a converter with a 256K buffer for $89 (#78870; #78920 without memory chips, $57.15). You need to wire the cable thus:

CoCo pin #	*DB-25 pin #*
2 (receive)	*20 (external ready)*
3 (ground)	*1 or 7 (ground)*
4 (transmit)	*3 (receive)*
1 (not used)	

Radio Shack sells a converter, and they may be found or ordered through local PC stores also.

I just received the Sept/Oct 96 issue and enjoyed very much the article by Allen Huffman on the EZ135 removeable drive on the CoCo. I hope to be able to get one by next spring. Currently I am using Jeff Vavasour's emulator (OS-9 only), but probably cannot get SCSISYS 2.2 from Northern Exposure.

Good luck with the endeavor to bring other platforms into the fold. I enjoy reading what everyone is doing and maybe they have some ideas or projects that could be ported to the CoCo.

Jerry Cantrill
2526-1A Potomac Hunt Lane
Richmond, VA 23233-1552

So far I haven't had much luck with the Atari crowd. I contacted a user group on the Internet and got a positive response, but haven't received any further word from them. I did run across a memory upgrade for the 520 and 1040 ST that I may print though. Maybe I'll hear from them by the time I you read this.... who knows? I may end up contacting the Commodore 64 crowd also.

Don't get me wrong. I'm fond of my CoCo and will continue supporting it, I even enjoy all the mail and ideas. Therer are enough subscriptions to continue for the next couple years or so too. I just want to ensure there will be continued support well into the future. And more subscriptions will ensure that.

Editor: The following letter was written in reference to someone pointing out that Lonnie Falk, former editor of "Rainbow Magazine", suggested CoCo users look to MS-DOS based machines when they outgrew the CoCo rather than 68K based OS-9 machines.

When you look at the typical CoCo user (the key here is typical) when the article was written and in the years previous to then you will come to understand that for the most part CoCo users did not want to learn OS-9, spend a lot of money on programs or do a lot of learning about how to use a computer. They, for the most part, wanted to plug 'n play. In that aspect a PC was a wiser upgrade choice. Many also wanted to play games and use the computer to do things like keep a checkbook, mailing list, word processor (with graphics), maybe a database or spreadsheet and not spend a lot of money. The PC had all of these programs for dirt cheap in shareware or Public Domain. Note that PCM (Falk's PC support magazine which he fulfilled Rainbow subscriptions with) even catered to the users that were interested in these inexpensive programs by having a shareware library that you could order programs from. A rather unique idea at the time.

The OSK machines never had a lot of cheap easy to use software to choose from and would also require a lot of learning to install and use these programs. Note that we had some very heated discussions about the use of installation programs or scripts under OS-9 back then with the OS-9 power users adamantly stating that scripts weren't needed or wanted and people like me asking why not do it anyhow. Who would it hurt? Since it was the power users that were for the most part writing the programs we saw very little software written that was easy to install for a novice, unlike the PC where you would type install and be prompted for just about every eventuality. The old argument went like this: "I don't have time to write a script to install a program. A user should know enough to be able to install the program. If I write a script somebody will still mess it up and waste my time on support of the installation script." Thus went the arguments and I gave up trying to persuade the programmers that they were pushing people towards PCs. In the long run we lost the majority of people that upgraded their computer systems because we didn't listen to them.

The MM1 is a perfect example. It was a decent machine when first introduced but MANY things were promised for it that never happened. It was intentionally crippled so that it could use a CM-8. I understand why but don't agree with the logic. It should have come with fully implemented serial ports, at the very least two fully functional ports and a mouse port. The windowing system was never finished and upgrades seemed to come out daily. This is not easy for users or programmers since one version would work one way and another would work differently. How many times did I see posts here asking which version of a module somebody was using when trying to get a program working? Compare this to Windows on a PC. You can count the upgrades to Windows for any given year on one hand, in fact in most years you could count them on one finger.

I guess I've gotten on a soapbox here again but Lonnie called it right. He was in business to make money and when the Rainbow couldn't pull its own weight he had to cut it loose. I know that somebody else could have kept it going, but if you are a good businessman the idea isn't to keep a loosing proposition going as long as possible, you use your resources to find areas that have potential for growth. Time and management are limited resources, most people forget that time is a resource when making decisions about closing a venture.

The CoCo community owes Frank Swygert a big thanks for publishing his magazine. I've subscribed to it from day one. He will be the first to admit that it takes a lot of time to put together an issue and that he isn't ever going to get rich doing it. Either he's a masochist or he is an enthusiast. Since I've never seen him wearing leather and I've seen the cars he loves I'd guess he's an enthusiast. Either way thanks again Frank for helping fill the void that the loss of the Rainbow created.

Carl Boll
root@chicoco2.chi.il.us

Well, I need to start this reply off by thanking Carl for the kudos! Yes, I'm a CoCo and OS-9 enthusiast, but must say most of my "CoCo time" goes into the magazine. I read and write more about the CoCo and OS-9 than I 'd ever have time to spend actually "playing" with them!

Carl has a definite point when speaking of hard to run OS-9 programs. Luckily that mold seems to be breaking. When Rick Ulland put "Patch OS-9" together he realized that an installation program was essential. So did Alan DeKok and the writers of NitrOS-9. Maybe others will see this also.

Floppy Drives Revisited

A follow-up of the July/August issue's disk drive article

Frank Swygert

CoCo Disk Drives

I would like to know what brand and model floppy drive to use for 80 track, 720K, 5.25" and 3.5" on the CoCo3. Thank you.

Dion VanIrvin
108 North 46th Street
Louisville, KY 40212

It doesn't take much to start an article. Sometimes just a simple question as the above note will do the trick. I'm not sure if the floppy drive article in the last issue spurred this note about brands of disk drives or not. I did look back at it and maybe I wasn't to clear on a few things. I must admit that the article was rather hastily put together. I'll try to remedy some of the shortcomings I found and add some additional information.

Which Drives Work?

ANY IBM/PC compatible drive up to 720K will work with the CoCo. It doesn't matter who makes them as long as they are PC standard drives. It doesn't make any sense at all to get a single sided drive anymore, so go ahead and get the 360K model if you need to replace a 5.25" drive. Sure, it only works as a single sided drive with DECB, but the drives no longer cost any more than the single sided models. My favorite source lists 360K drives for $9.95.

There is a modification for the older Tandy 12 volt floppy controllers that allow use of 1.4M drives. This is an extensive operation requiring a good bit of soldering. The old controllers are also hard to find. This was detailed in the March/April 1996 issue for those hardware hackers wishing to make the modification. It would be best for most users to stick with using 720K drives.

720K 5.25" Drives

There were several manufacturers of 720K 5.25" drives. I don't have any model numbers. A lot of CP/M machines used them. The only IBM/PC compatible to use them was the Tandy 2000. These drives are electrically identical to the 720K 3.5" drives. There is really no reason to try to find a drive like this unless you already have a lot of 720K 5.25" disks with information you need to retrieve. This type drive hasn't been made in years now. Although you can "double step" the drive to make it read a standard 35 or 40 track OS-9 disk, you can't write to such a disk unless it is freshly formatted in the 720K drive. You can't format it in a 35/40 track drive then write to it, or format over such a formatted disk. The reason is that the heads of the smaller capacity drive will try to read both the new and old information at the same time, resulting in a very confused computer. This is sometimes a problem with PCs when reading and writing 360K disks on a 1.2M floppy. The general rule is to format a fresh, never before used disk at 360K on the 1.2M (or 720K) drive and write only to that disk with the high capacity drive, NEVER the 360K drive. And never write to a 360K disk formatted on a true 360K drive with a higher capacity drive.

720K 3.5" Drives

The best recommendation at this time is to get a 720K 3.5" drive. There are still a few made, and a lot on the surplus market (there used to be a lot of the 720K 5.25" drives available surplus, which is why early CoCo OS-9 users adopted them, but the supply has dried up). You can expect these drives to be on the surplus market for another three to five years. If you have one of the 5.25" 720K drives, get a 3.5" and transfer your data before your drive fails or the disks go bad. Quality disks will usually hold data for four to five years when stored in an air conditioned, heated environment. The higher the quality and enviromental control, the longer your data is safe.

Using 1.4M Drives as 720K

Most 3.5" floppy drives rotate at 300RPM and can be safely used as 720K drives. The media type switch controls drive speed selection of those that operate at 360RPM for 1.4M operation (only a very few switch speeds). There should be no problems when using 720K media.

1.2M drives are a different story. Almost all operate at 360RPM when in high density mode. Many always operate at the higher RPM rate, making them unusable as 720K drives with a CoCo floppy controller. Some 1.2M drives are jumper selectable for 720K only operation.

Later in this article the settings for some common 1.2M drives are discussed by Gene Heskett. No information is available for other drive models at this time. If you discover the proper settings, please forwward to this magazine.

Combo 5.25"/3.5" Drives

The combo drives popular in today's PCs can also be used in a CoCo. They must be configured similarly to the 1.2M drives. A few rotate at 300 RPM, others are dual speed. You will have to contact the manufacturer or dealer to get the rotational speed. Jumpers may have to be set to force the drive into low density mode and/or low speed operation. See the instructions for 1.2M drives for tips. You may also be able to contact the manufacturer for jumper settings.

You will need a cable with a "full set of teeth" for dual size drives. Most CoCo 3 setups will have the proper connectors, but an older drive cable may have missing teeth. These drives usually come configured as drive 0 (3.5") and drive 1 (5.25"). Most have jumpers to switch the two, but not to change the drive numbers. A 3.5" type drive connector may be required. These are available from Radio Shack and can be crimped on your existing cable.

Drive Sources

There are several sources for the 3.5" 720K drives. My favorite is Hi-Tech Component Distributors (800-406-1275; 59 S. LaPatera Lane, Goleta, CA 93117; www.internet-cafe.com/ hi-tech/). They have a lot of surplus drives, including SCSI hard drives and all sizes floppies from 360K 5.25" to 2.8M 3.5". The following sources listed used/surplus computer parts, but I have not specifically done business with them or checked them out:

Dallas Computer Parts
1135 E. Plano Parkway, Ste. 2
Plano, TX 75074
214-422-1580

Computer Again
132 Garza Boulevard
Farmingdale, NY 11735
516-845-7502

Don't forget to try local computer stores, especially those who work mostly with businesses. They do a lot of upgrades and may have some used drives around that they would sell. The average price for a 3.5" 720K drive is $20-$30 and $5-$15 for 360K drives, depending as much on stock-in-hand as brand and condition (new, unused, or used... unused is usually one that was pulled from an unsold older system, but never used... effectively a "new" drive, but can't be sold as such because it was installed in a system or removed from the original packing).

continued on page 9

The Need for Speed...

A. Hsiao

Speed up CoCo with a crystal change!

I created this article in order to explain how I (not necessarily you) got my CoCo 3 running with a 38 MHz crystal, effectively boosting the clock speed of my machine from 1.78 MHz to 2.375 MHz (a noticeable improvement!). Following is the information in "step by step" format. PLEASE don't do this if you're not experienced at least a little bit with a soldering iron and electronic circuitry. It requires opening both your CoCo3 and your monitor, and there's no gurantee it will work so you may have to reverse the process.

BENEFITS

The faster crystal in a CC3 will noticeably improve speed, not only under OS-9, but under DECB as well. The speed increase will be relative in everything, not just math, text scrolling, or disk access.

While at first the increase from 1.78 to 2.375 (with a 38 MHz crystal) sounds small in number, when you calculate it, it comes to a 33.5% speed increase, twice the increase boasted by PowerBoost or NitrOS9. When coupled WITH NitrOS9 or PowerBoost (as I have done), your CoCo will be approximately 50% faster than a stock Color Computer 3.

DRAWBACKS

1: You are using the computer and monitor at speeds for which they were not originally designed. While I've been using the 38 MHz crystal for some time now, things run a little hotter, and the GIME obviously struggles sometimes (i.e. it momentarily stops generating sync every once in a while, and runs very warm). *Editor's note: A heatsink may help this. Radio Shack should have one.*

2: Software's timing will be off. Software that uses the fixed rate of the original crystal for internal workings will either not work at all, or will function differently. While there isn't much of this software, it does exist. OS-9 is one of these, but there's a fix which will be described later.

3: You will have to replace your 150 or 120 nanosecond DRAMs with 100 nanosecond or better. Users of the Tandy 512k upgrade are shut out from using crystals over 32 MHz, because the board is mounted face down and generates so much heat at higher speeds that the machine crashes very quickly. *Editor's note: An external power supply, such as a PC supply, will help here. If you don't want to use a PC type supply, take your transformer out and make a longer cord between the transformer and the CoCo. You'd be surprised at how much heat this removes, and you can use the extra room for a small fan.*

4: You may not be able to get a display. Monitors are designed to sync at specific rates. When you alter your crystal value, your monitor will be expecting a different rate, and thus, will display data incorrectly. Thus, you will in most cases be required to open up the actual monitor and adjust internal workings.

WHAT YOU WILL NEED

* A crystal of known value
* Soldering Iron, Solder Sucker, and Solder
* Tools to open the CoCo case and Monitor housing (in most cases)
* OS9 Users: The Eddie Kuns CLOCK module (edition #9). This is needed because it allows you to easily change the value of the clock rate. It's also much better than the stock OS-9 clock module.
* The guts to open up your CoCo and monitor and modify them!

STEP 1: THE CRYSTAL

Crystals come in almost an infinite number of values, from less than 1 MHz to almost a hundred. The idea is to choose the highest value of crystal that your system will function with. Here are the experiments from my system. It should be noted that I have a 2 MHz rated 6309 (63B09) installed in my system, I have not attempted this with a 2 MHz 6809 (68B09). This should not be a problem, but the 68B09 will run warmer than the CMOS 6309. A heat sink may be in order for the 6809. Also note that while this is the way my system responded, yours may be entirely different, as CoCo hardware is noted for inconsistency.

I have included values slower than the original crystal just because I figured that while I was playing around with speed, I may as well get all the information possible! The adjustments made will be detailed later in this document.

STEP 2: INSTALLATION

Installing the crystal is very simple.

1: Open your CC3 case and locate the crystal. It is a silver cylinder under the keyboard at the edge of the motherboard and is marked "KDS 28.63636" (or a similar value for PAL machines).

2: With your soldering iron and solder sucker, desolder crystal and remove it.

3: Insert the new crystal and solder in. It's as simple as that.

STEP 3: MOMENT OF TRUTH

Hook your CoCo and monitor up. Turn on the monitor first, and give it a few moments to warm up. This is so that you can see what happens immediately after you turn your machine on and can turn it off again quickly if it's bad! After you've given the monitor 10 or 15 seconds, turn the CoCo on. If all appears to be working fine, then your hardware will require no further adjustment, and you can go pour yourself a cup of coffee and stare at your faster CoCo! OTHERWISE, check the following:

Value	CPU	MHz	MultiSync	CM-8	VM-5
4.000	No func	0.2	—	—	—
8.000	Normal	0.5	Bad Disp	No Sync	No Sync
18.375	Normal	1.1	Display	Adjust	No Sync
24.000	No func	1.5	—	—	—
*28.636	Normal	1.8	Display	Display	Display
30.000	Normal	1.9	Display	Adjust	Adjust
31.117	Normal	1.9	Display	Adjust	Adjust
33.880	Normal	2.1	Display	Adjust	Adjust
38.000	Normal	2.4	Adjust	Adjust	No Sync
40.000	No func	2.5	—	—	—
45.148	No func	2.8	—	—	—
49.875	No func	3.1	—	—	—

* This is the speed of the standard CC3 crystal, and was included for reference.

No function = CPU did not function at all
Normal = CPU functioned normally and booted OS9
Bad Disp = Display was present but small and shaky
No Sync = Monitor could not be adjusted to display
Adjust = Monitor displayed properly, with internal adjustments
Display = Monitor displayed properly and require no internal adjustments
CM8 = Tandy CM8 Monitor.
MultiSync = Sony MultiSync monitor.
VM5 = Tandy VM5 monochrome monitor.

1: If the screen flashes garbage and then goes blank, chances are your CPU is not generating a video signal, and your crystal is either too fast or too slow for the CPU to function right.

2: If the screen is just blank and flashes nothing, then your crystal is probably bad, and you should try another one.

3: If the screen displays a permenant barrage of "garbage", then your CPU is probably functioning right, and the monitor is not displaying.

Try the following (assuming you've gone into RS-DOS):

Type CLS1 and press ENTER. The "garbage" should change it's general color. Try it again with CLS2 or CLS3. If the garbage is changing colors then you have a working CPU and a monitor not displaying properly. If CLSX does NOT work, then shut your CoCo off, this crystal is either too fast or too slow for your CPU.

STEP 4: MONITOR ADJUSTMENTS

There are a number of monitors popular with CoCo users. I will only list two here specifically, along with some general instructions.

CM-8:

The CM-8 monitor is a decent monitor and can be adjusted to display all the way up to 38 MHz (crystal).

1: Open up the CM-8 monitor case. This entails removing six screws: Two at the top rear, two at the bottom rear, and two in the rear panel which the cables come out of.

2: Once open, you will see lots of stuff, most of which should be ignored. What you are looking for is a small "pot" that is (looking from the top front) located on the left edge of the circuit board, and will be exposed without requiring you to pull the circuit board all the way out.

3: There should be three pots located here, and the one that you will want to play with is labeled "H. Hold". With the CoCo on, take a small screwdriver and adjust this pot in either direction until the "garbage" turns into a spinning display (the kind that requires V-Hold adjustment).

4: After getting the spinning display, adjust your V-Hold knob on the front panel until the display locks into place.

5: At this point, you may want to play around with the pot next to it called V.Size until you get the vertical size up to a normal level (easiest done in Width 80). Once this is done, you can close up your monitor, and you're done for the hardware!

SONY MULTISYNC:

The Sony monitor had an interesting display problem. At higher crystal values, the display would be readable, but hourglass-shaped when the CoCo was turned on. This was rectified as follows:

1: Open up the sony monitor. The back cover is held on by four screws, two on top and two on the bottom. Once the screws are loosened, take the cover completely off.

2: With the CoCo on, locate a hole on the image board (the most complex and largest one), marked "H.Sync". It is near the front, and about halfway up.

3: Insert a small, straight-edged screwdriver into the hole and rotate until the hourglass suddenly pops out to a normal display. V.Sync, as you will see, automatically adjusts (it's a MultiSync monitor...).

4: Close the monitor, and you're done with the hardware.

— OTHER MONITORS —

Generally, there are two values you will need to adjust. On both MultiSync and other monitors, you will need to adjust the horizontal sync rate, commonly labeled as "H.Sync", "H.Hold", or "Horiz. S.". On NON-MultiSync monitors ONLY, you will also need to adjust "V.Sync", "V.Hold", or "Vert. S.", otherwise known as the vertical sync rate.

STEP 5: SOFTWARE INSTALLATION

At this point, exclusive RS-DOS users can stop. OS-9 users, however, will experience a little bit of trouble. Because OS-9 derives its CLOCK timing from the crystal oscillator's fixed rate, the system clock, with a faster crystal, will run fast (the 38 MHz crystal causes about a 2 minute gain every five real minutes!). The remedy for this is the CLOCK module by Eddie Kuns, edition #9. This is a full-fledged GIME toggle clock, faster than the stock OS-9 clock, and with versions for Disto and Burke and Burke RTC's, so have no fear.

First thing you will need to do is go through the standard procedure for creating and OS-9 bootfile (refer to the OS-9 manual, and to the millions of documents telling you how to do so), replacing the standard Clock.60Hz with your appropriate version of the Clock #9 module (depending on whether you're using a B&B, Disto, or software clock).

Boot with this new clock, and then, using modpatch (or for us Burke and Burke people, EZGen), you will need to chang the clock module as follows:

1: Divide the value of the original crystal (28.63636) by 60.

2: Then, divide the value of your new crystal with this value.

3: Round it off to the nearest decimal number, and, using a calculator or computerese friend, convert it to hex.

4: Change the value at location $7B from $3C to the new value.

5: Do the cobbler thing, or for EZGeners or KWIKGeners, save the new clock module to your bootfile (or write the bootfile).

YOU ARE NOW DONE! Enjoy the added speed.

END NOTES:

I was able to test so many crystals because I installed ALLIGATOR CLIPS into the solder pads that held the original crystal. Don't solder and desolder this much on your motherboard, as you will certainly muck it up!

My system has the following hardware, which all function fine with a 38MHz crystal and the OS-9 mods installed:

CoCo3 Motherboard, PBJ 512k board with 100ns 41256 DRAMs

MPI, Upgraded for CC3 operation

Disto Super-Controller II

Disto 3-in-1 board

Burke and Burke Hard Disk Interface

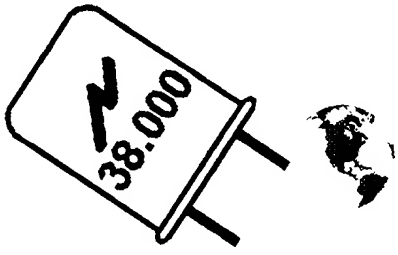
Western Digital WD1002-27X RLL hard disk controller

2 Tandy ACIA paks, one modified to map out at $FF6C

CM-8 Monitor

All hard drives (SEAGATE ST4144R and NEC D5146) and floppy drives function properly, as well as do the modem, printer, and Puppo keyboard interface. Since these peripherals function correctly, most others should also.

The author may be reached via e-mail (JZR@onlybbs.via.mind.org) or through this magazine for further comment.



FARNA Systems

Your most complete source for Color Computer and OS-9 information!

Post Office Box 321
Warner Robins, GA 31099
Phone: 912-328-7859
E-mail: dsrtfox@delphi.com

ADD $3 S&H, $4 CANADA, $10 OVERSEAS

BOOKS:

Mastering OS-9 - $30.00
Completely steps one through learning all aspects of OS-9 on the Color Computer. Easy to follow instructions and tutorials. With a disk full of added utilities and software!

Tandy's Little Wonder - $25.00
History, tech info, hacks, schematics, repairs,... almost EVERYTHING available for the Color Computer! A MUST HAVE for ALL CoCo afficiandos, both new and old!!!

Quick Reference Guides
Handy little books contain the most referenced info in easy to find format. Size makes them unobtrusive on your desk. Command syntax, error codes, system calls, etc.
CoCo OS-9 Level II : $5.00
OS-9/68000 : $7.00

Complete Disto Schematic set: $15
Complete set of all Disto product schematics. Great to have... needed for repairs!

"A Full Turn of the Screw": $20
Lots of CoCo info, projects, and tutorials,. by Tony DiStefano

"Inside Disto's 2 Meg Kit" : $10
Schematics and explanation of how the 2 meg CoCo 3 upgrade works.

SOFTWARE:

CoCo Family Recorder: Best genealogy record keeper EVER for the CoCo! Requires CoCo3, two drives (40 track for OS-9) and 80 cols.
DECB: $15.00 OS-9: $20.00

DigiTech Pro: $10.00
Add sounds to your BASIC and M/L programs! Very easy to use. Requires user to make a simple cable for sound input through a joystick port. Requires CoCo3, DECB, 512K.

ADOS: Most respected enhancement for DECB! Double sided drives, 40/80 tracks, fast formats, many extra and enhanced commands!
Original (CoCo 1/2/3) : $10.00
ADOS 3 (CoCo 3 only) : $20.00
Extended ADOS 3 (CoCo 3 only, requires ADOS 3, support for 512K-2MB, RAM drives, 40/80 track drives mixed) : $30.00
ADOS 3/EADOS 3 Combo: $40.00

Pixel Blaster - $12.00
High speed graphics tools for CoCo 3 OS-9 Level II. Easily speed up performance of your graphics programs! Designed especially for game programmmers!

Patch OS-9 - $7.00
Latest versions of all popular utils and new commands with complete documentation. Auto-installer requires 2 40T DS drives (one may be larger).

NEW ITEMS!!!

FARNA Systems is pleased to announce that we are now distributors of the following, formerly from Northern Exposure! Note: If you never received your order from NX, send a copy of yourcancelled check along with $5 to cover S&H and I'll fill the order!

NitrOS-9 : $35.00
A complete rewrite of OS-9 Level II that takes advantage of all features of Hitachi's 6309 processor. Easy install script! 6309 required.

TuneUp : $20.00
If you don't have a 6309, you can still take advantage of some of the Nitro software technology! Many OS-9 Level II modules rewritten for improved speed with the stock 6809!

Thexder OS-9
Shanghai OS-9 : $25.00 each
Transfers your ROM Pack game code to an OS-9 disk! Please send manual or ROM Pack to verify ownership of original.

Rusty : $20.00
Launch DECB programs from OS-9! Allows loading of some programs from hard drive!

FARNA Systems AT306 Based Computers

Complete computer systems based on the AT306 board from Kreider Electronics. Systems are completely setup and ready to go, just add a monitor (or we can supply that too)!

Both systems include:

16 bit PC/AT I/O bus with five slots
MC68306 CPU at 16.67MHz
4 30 pin SIMM sockets
IDE Hard Drive Interface
1.4MB Floppy Drive
Two 16 byte fast serial ports (up to 115K baud)
Bi-directional parallel printer port
Real-time clock
PC/AT keyboard
Desktop Case and Power Supply
BASIC (resembles Microsoft BASIC)
MGR Graphical Windowing Environment with full documentation
"Personal" OS-9/68000 Vr 3.0 (Industrial with RBF)
Drivers for Tseng W32i and Trident 8900 VGA cards
Drivers for Future Domain 1680 and Adaptec AAH15xx SCSI cards

Many other utilities and tools

FARNA-11122 Includes:
2MB RAM
200MB Hard Drive*
Trident 8900 1MB Video Card
$960.75

FARNA-11225 Includes:
2MB RAM
500MB Hard Drive*
Tseng W32i 1MB Video Card
$1114.47

This is the SMALLEST amount of formatted space (most manufacturers specify UNFORMATTED CAPACITY). Prices fluctuate - we get you the largest drive possible for the money allotted!

Call for a quote on different configurations and components.
Warranty is 90 days for labor & setup, components limited to manufacturers warranty.

Microware Programmers Package -
Licensed copies of Microware C compiler, Assembler, Debugger, Basic, and many other tools!
***With system purchase:* $65.00 *Without system:* $85.00**

More software will be listed later!

Microware Notes

Boisy Pitre

An educated opinion on Microware's OS-9 policies

After reading another OS-9 community member's assessment of Microware's seeming disinterest in his club's efforts to promote OS-9, I wanted to add a few comments. As always, my opinions are mine only, and not those of my employer (Microware). I have given the editor permission to print my opinion here.

It comes as no surprise to many here that the OS-9 hobbyist market is all but gone. There has been talk of a new OS-9 system recently, and while this is all well and good, I really don't see much of a sustained profit to be made from these types of endeavours. Moreover, I don't envision a revival of the OS-9 hobbyist market unless substantial revenue from an outside source would be invested.

Microware, as a public company, is focusing on markets that have the potential to create revenue for itself and its shareholders. We're not talking hobbyist computer systems here; we are talking mass-produced consumer electronics with possibly billions of potential dollars at stake.

Case in point: I'm looking at an article on my desk dated August 29, 1996 in the Des Moines Register titled "Microware software to power 'Net TVs." It goes on to discuss how Mitsubishi is coming out with DiamondWeb, a 40" TV set which will have David/OS-9 and a web browser built-in. And this is just one of several markets OS-9 has penetrated recently.

Ironically, isn't this what OS-9 users have wanted all along? For OS-9 to be widely accepted and used? It may be wrapped around a name like "David" or "CD-i," but of course, it's OS-9. And our experience with CoCo 3's, MM/1's, System IV/V's or any other OS-9 machine is enough to catapult some of us into the OS-9 "labor force."

These machines will always be a part of OS-9's legacy, no doubt about it. And because of them, several have gone on to successful careers at Microware and other companies doing OS-9 work. I can count at least 8 people off the top of my head. Some of them are subscribers.

OS-9 is maturing and changing. As OS-9 users and fans, we should recognize th evolution and be appreciative of the fact that in some small way, we helped contribute to OS-9's current success by buying the product, using it and promoting it. We shouldn't be bitter because our market (hobbyist) is no longer a viable one. We must accept what is, and adapt to what becomes.

Regarding the comment "Microwares lack of interest, in this type of venture, does not say much for the company"... One really needs to get informed about the industry and where it has been heading before making such an unwarranted statement. Microware has targeted and is acquiring a potentially huge market. As a result, OS-9 will be on devices that we could have only dreamed of 8 or 10 years ago.

More letters...

I agree with your reasons for possibly supporting the Atari, especially since there are no Atari based magazines being printed here in the US. The only Atari magazine from North America is "Current Notes" which is out of Canada. Given some of the Atari clones coming from Europe, I think that there will be Atarians around for quite some time.

Since I am a new reader to your magazine and realitively new to the CoCo list, I was curious as to what the MGR system is? Is it one of the 306 based systems such as the WPC 306 or is it something else?

Looking forward to the next issue and if I can find the time, I'll try and write and article or two. Thanks again and keep up the good work.

Jim Cox
jimc@amc.com

As I mentioned to Jerry Cantrill in another letter, there hasn't been much response from the Atari crowd so far. MGR stands for "manager". It does, as you guessed, run on the AT306 based machines (from BlackHawk, Wittman, and FARNA). MGR is a graphical interface similar to Multi-Vue, only better. It is more like MicroSoft Windows or G-Windows, but not expensive. The version that comes with the AT306 machines was written in 1988 and is derived from a public domain version. There is a commercial version available for 680x0 machines that is popular in Europe. If the OSK machines really caught on, this could be an easy upgrade path that would allow use of all software written for the PD version.

The Syquest articles were great and some of the best I've seen. But where do you get a Ken-Ton SCSI interface? And which version of the drive do you get... Mac, PC, or parallel?

Ray H Baumiller, Jr.
Rt 4 Box 356
Everett, PA 15537-9717

Ray, you would need the SCSI version of the Syquest EZ135 or newer EZFlyer (230MB). Unfortunately, the Ken-Ton interface is no longer avaialble. Try to find a used SCSI interface (any make will do) or a Disto hard drive interface. I am working on a source for a few SCSI interfaces. I would also be willing to publish plans fro a SCSI interface is anyone out there has one...

I am writing this letter to let you know how much I enjoy "the world of 68' micros". Included with this letter is my renewal.

In your disk drive article there were some errors. Corrections are enclosed. If anyone is looking for 3.5" 720K drives, All Electronics (1-800-826-5432) has new Sony drives for $14.00 plus shipping. These particular drives have a larger than normal faceplate, but can be easily modified to fit a standard case. I am using one on my CoCo and another on my PC/XT clone.

William F. Brown
Box 124
Wysox, PA 18854-0124

William, I didn't print all your corrections as I used them in a follow-up article in this issue. Thanks for setting me straight on some issues and clearing a few others up.

Floppy Drives Revisited

continued from page 4

Setting 1.2 Drives for 720K Operation

Gene Heskett & "Phil"

The following will describe how to use a Panasonic (or Matsushita or Shugart, they are all made by the same company) JU475 drive (1.2 meg) as if it was a 720K drive.

The trick is how to force the drive to rotate at 300 RPM when in low density mode, and how to force it into low density mode after making it into a dual speed drive. MOST 1.2 meg drives normally rotate ONLY at 360 RPM, but many of them have jumpers to allow them to rotate at 300 RPM when in low density mode.

The BX and CX (if present) jumpers control whether the drive is dual speed or not. Mess with them to make it work as a dual (360 / 300 rpm) speed drive instead of the default of 360 rmp only.

There are two pins on the drive edge connector not used by 360K drives that are sometimes used on high density (1.2, 1.4 meg) drives. These two pins are as follows:

Pin 2: Often used as a DENSITY SELECT. I'd start by forcing this one high (applying +5V) or low (grounding) to make that 1.2 meg drive (after setting jumpers as per my note) act like a 720K 80 track low density drive. On the JU-475-3A36, the BX jumper seemed to enable Pin 2, which needed to be brought low to slow it down.

Pin 34: used either as a disk change line (drive to computer) or a drive ready line (drive to computer). Or not used on some XT computers, and on all CoCo's, of course (usually used as disk change on PC compatibles).

Panasonic JU 475-3 : Jumper BX shorted (No CX on this drive). I also had to move a jumper on 'OP' to 'M1', the next adjacent set of pins in that row, the last set on the right or that row, looking at the pcb from the rear. This was finally determined by hooking up the power lead, going down the connector with a grounded jumper until the motor started, then listening to the motor as the rest of the connector was probed with another grounded lead. On pin 2 of the edge connector, the motor slowed down. Tracing that with a beeper meter, I found the jumper on 'OP' was connected to it, and that the front, away from the edge connector pin of M1 was common to it. The rear, closer to the connector pin of M1 was grounded. So I moved the jumper, isolating the card edge connector pin 2, and effectively putting an internal logic zero at that point forever, or until I move the jumper.

I figured that was it, fired off 'ded /f1@' to see if it stepped correctly. The first track read ok, with DD.FMT($10)=03 but changed to $07 and PD.DNS(offset $43)=03 already, disk was write protected, removed tape, re-wrote data. Hmm, can't get past sector $23, doing the head cleaner jerks. Re-ran 'ded /f1@', fixed DD.FMT up to $07 again, it was still $03! This time it works! Now to put it all together, fix the bootfile and use it for a while. Again, thank you very much. The BX jumper as a starting point was the magic bullet.

TEAC FD-55GFR-1XX

Six jumpers have been installed:

FG : Connect ground to chassis.

D1 : my drive select.

DC : Select output signal for pin #34.

DC-ON/RY-OFF : disk change.

DC-OFF/RY-ON : ready.

(Never set DC and RY on)

LG : Select density mode, pin #2 from controller. High (+5v) = LG-OFF (high density). Low (grounded) = LG-ON (ow density 720K)

#2 low (0v) :LG-OFF : Low density 720KB :LG-ON : High density 1.2MB

I,IS : Select speed modes

I-OFF / IS-OFF : Single high speed selected 360RPM

I-ON / IS-ON : Dual speed selected 300/360RPM. You will need to set the drive to dual speed mode. Speed will automatically be changed when drive density changes on dual speed selections.

At this point, simply dmode a new descriptor:

```
dmode /d1 stp=03 cyl=50 sid=02 dns=03
```

Be careful not to mistake the "I" for an "H" on the jumpers!.

This info was obtained from TEAC's automated FAX system at (213) 727-7629. This is in Montebello, CA (near LA). Much technical info is available. File #1002 is a list of available documents.

Some Corrections...

I made a few errors in the last article. Luckily, a couple readers corrected me. I had stated that all high density drives rotate at 360 rpm. This isn't true, all 360K, 720K, and 1.4M drives rotate at 300 rpm. Only the 1.2M drive rotates at 360 rpm. I also stated that high density disks were more sensitive than low density. I had this backwards! The 1.2M drive head emits a stronger magnetic field to write data.

I had mentioned that high-density drives store 512 bytes per sector, and the EHD or XHD (2.8Meg) drives store 1024 bytes per sector... not true. Bytes per sector on a floppy are not a function of the drive... they are a function of the OS. On a 360K disk, CoCo's store 18 sectors per track (256 bytes per sector), but IBMs store only 9 sectors per track (512 bytes per sector). IBM drives have the following specs:

5.25" 160K - 512 bytes/sector, 9 sectors/track, 35 Track, 1 Side, 48TPI

5.25" 180K - 512 bytes/sector, 9 sectors/track, 40 Track, 1 Side, 48TPI

5.25" 320K - 512 bytes/sector, 9 sectors/track, 35 Track, 2 Sides, 48TPI

5.25" 360K - 512 bytes/sector, 9 sectors/track, 40 Track, 2 Sides, 48TPI

5.25" 720K - 512 bytes/sector, 9 sectors/track, 80 Track, 2 Sides, 96TPI

3.5" 720K - 512 bytes/sector, 9 sectors/track, 80 Track, 2 Sides, 135TPI

5.25" 1.2M - 512 bytes/sector, 15 sectors/track, 80 Track, 2 Sides, 96TPI

3.5" 1.44M - 512 bytes/sector, 18 sectors/track, 80 Track, 2 Sides, 135TPI

3.5" 2.88M - 512 bytes/sector, 36 sectors/track, 80 Track, 2 Sides, 135TPI

IBM's always use 512 bytes per sector. It's bytes per track would give you density. Also, there are 3 track densities, not 2. All 3.5" floppies are 135TPI.

2.8M drives do *not* use the same media as 1.4M drives! They require a special disk with an even higher flux density rating (have never seen the official spec, tho) and still cost roughly $3.00 per disk! Between this cost, and the cost of the drives still hover around $70.00 is one of the main reasons they never caught on... the cost per megabyte was way out of sight, and no-one wanted to pay that much!

I'm sorry if my misinformation caused any confusion. I'll try to do better research from now on and not get in a rush to put out interesting articles. And thanks to William Brown and Roger Merchberger for correcting me!

The Zen of Color Computer Programming

Chet Simpson

A CoCo Cross Development System

Editor: In the last issue, we brought Chris Dekker's Basic09 programming series to a close. In this issue, Chet Simpson starts to show us how to get the most from CoCo's native Disk BASIC system. CoCo still has a lot of life left in it, and DECB is still the best way to directly manipulate the hardware. If you are programming telecommunications applications or games, DECB may be the best way to go...

The Challenge...

Recently, a long-time friend of mine gave me what he calls the ultimate "Challenge," to write a full screen, scrolling platform type of game for the Color Computer 3. My friend is Eric Crichlow, author of Gold Runner 2000 for the MM/1. Quite some time ago, he made a comment that Gold Runner 2000, or GR2k as we all call it, was impossible to do on the CoCo. I personally thought that he made this statement in haste and gave him a tongue lashing, explaining that the CoCo could do it, and do it better! Thus "The Challenge" was born!

After the initial excitement of The Challenge subsided, I sat down and looked at the typical development process used on the CoCo under DECB (RS-DOS). Generally it consists of loading up an assembler, making changes to source files, assembling them into an executable and loading it to see if it worked. This is generally OK during the initial stages of development, but because the source files grow so quickly, so does the assembling time. If I am working on some obscure time-critical IRQ driven routine, I may find myself rebooting and reassembling several times a day. If I am writing self modifying code that is constantly changing itself, I can figure on not sleeping that week.

Because of the amount of time assembling and loading consume, I decided to look at other alternatives. Since most of my development efforts for the past couple of years were on 32bit based computers, I wanted to use my current "development" machine. This formed the initial designs for the ultimate 6809/6309 cross-development system (CDS) [targeted for CoCo OS independence].

What is a CDS you ask? It is a set of programming tools that allow you to develop software on one operating system (known as the host) for a computer with a different operating system (known as the target). This is definitely not a new concept and has been in use for several years. Even many of today's most popular games are being developed on higher-end systems such as SUN and NeXT workstations. One good example is Doom, which is possibly the most popular game to date, was written in 99% ANSI-C and the rest in assembly all on a NeXT workstation.

The primary goal of the CDS is to write the assembly code on a "host" machine, create the executable, transfer it to the CoCo and run it. In order to do all of these things, several tools are needed. Since there is an abundance of text editors this is was not a major concern.

Creating the executable was another story. In order to do this, a low-cost assembler was needed. How about one that is free? Many years ago (before the CoCo 3) Motorola released the source code to a set of "Freeware" assemblers for their entire line of 8 bit microprocessors. Unfortunately, the assemblers are minimal and lack several features. The only alternatives were to write one from scratch or modify the existing 6809 assembler. I opted for the latter and went to work adding features such as including external source files, simple macros, 6309 mnemonics and conditional assembly. In its current form, the assembler is virtually bug free and will assemble both EDT/ASM (from CerComp) and EDTASM (Tandy/Radio Shack) source code.

The next step, and probably one of the most important aspects of the CDS, was deciding how to transfer the executable from the host machine to the CoCo. I could load a terminal program each time, download the executable from the host machine, restart the CoCo and test it, but that's a couple of more steps than I was willing to take. I wanted to be able to hit a single key, transfer and save several files all at the same time. In order to accomplish this, I was going to have to write a transfer system for both the host and target machines.

Since this was to be the "core" of the system, I designed every aspect of it with integration, usability and modularity in mind. The final result was a lot more than I had originally planned. This "core" system now includes support for serial I/O sequential and sector based disk I/O, keyboard input and hardware text display. Along with the hardware support, I also have a command line prompt that allows me to do simple commands. As a bonus, it also allows other executables to use its I/O capabilities when they run.

The CDS is a non-interrupt driven system. What this means is that although interrupts are not disabled, the CDS does not use them. All input from the keyboard and serial ports are polled. It was created this way primarily to allow the developer to have complete control over how the interrupts are used without too much of a conflict with the CDS. Although convenient for the developer, this does pose some problems. First, any functions that do disk I/O such as reading or writing sectors cause a loss of interrupt capabilities for a very short time. This is primarily due to the standard disk controllers not having what I consider a reliable interface.

During the design stage, I made several notes regarding what was needed for a command line shell. Unfortunately, time was tight and I needed something simple and quick. The final result was a simple input line with a limited number of hard coded commands. While this strays far from the original concept of modularity, it works for the time being.

continued on page 13

HawkSoft

28456 S.R. 2, New Carlisle, IN 46552
219-654-7080 eves & ends MO, Check, COD; US Funds
Shipping included for US, Canada, & Mexico

MM/1 Products (OS-9/68000)

CDF $50.00 - CD-ROM File Manager! Unlock a wealth of files on CD with the MM/1! Read most text and some graphics from MS-DOS type CDs.

VCDP $50.00 - New Virtual CD Player allows you to play audio CDs on your MM/1! Graphical interface emulates a physical CD player. Requires SCSI interface and NEC CD-ROM drive.

KLOCK $20.00 - Optional Cuckoo on the hour and half hour!! Continuosly displays the digital time and date on the /term screen or on all open screens. Requires I/O board, I/O cable, audio cable, and speakers.

WAVES vr 1.5 $30.00 - Now supports 8SVX and WAV files. Allows you to save and play all or any part of a sound file. Merge files or split into pieces. Record, edit, and save files; change playback/record speed. Convert mono to stereo and vice-versa! Record and play requires I/O board, cable, and audio equipment.

MM/1 SOUND CABLE $10.00 - Connects MM/1 sound port to stereo equipment for recording and palyback.

GNOP $5.00 - Award winning version of PONG(tm) exclusively for the MM/1. You'll fo crazytrying to beat the clock and keep that @#$%& ball in line! Professional pongists everywhere swear by (at) it! Requires MM/1, mouse, and lots of patience.

CoCo Products (DECB)

HOME CONTROL $20.00 - Put your old TRS-80 Color Computer Plug n' Power controller back on the job with your CoCo3! Control up to 256 modules, 99 events! Compatible with X-10 modules.

HI & LO RES JOYSTICK ADAPTER $27.00 - Tandy Hi-Res adapter or no adapter at the flick of a switch! No more plug and unplugging of the joystick!

KEYBOARD CABLE $25.00 - Five foot extender cable for CoCo 2 and 3. Custom lengths available.

MYDOS $15.00 - Customizable, EPROMable DECB enhancement. The commands and options Tandy left out! Supports double sided and 40 track drives, 6ms disk access, set CMP or RGB palettes on power-up, come up in any screen size, Speech and Sound Cartridge support, point and click mouse directory, and MORE OPTIONS than you can shake a stick at! Requires CoCo3 and DECB 2.1.

DOMINATION $18.00 - Multi-Player strategy game. Battle other players armies to take control of the planet. Play on a hi-res map. Become a Planet-Lord today! Requires CoCo3, disk drive, and joystick or mouse.

SMALL GRAFX ETC.

"Y" and "TRI" cables. Special 40 pin male/female end connectors, priced EACH CONNECTOR -	$6.50
Rainbow 40 wire ribbon cable, per foot -	$1.00
Hitachi 63C09E CPU and socket -	$13.00
512K Upgrades, with RAM chips -	$72.00
MPI Upgrades	
For all large MPIs (PAL chip) -	$10.00
For all small MPIs (satellite board) -	$10.00
Serial to Parallel Convertor with 64K buffer, cables, and external power supply -	$50.00
2400 baud Hayes compatible external modems -	$40.00

ADD $2.00 S&H TO EACH ORDER

SERVICE, PARTS, & HARD TO FIND SOFTWARE WITH COMPLETE DOCUMENTATION AVAILABLE. INKS & REFILL KITS FOR CGP-220, CANON, & HP INK JET PRINTERS, RIBBONS & vr. 6 EPROM FOR CGP-220 PRINTER (BOLD MODE), CUSTOM COLOR PRINTING.

Terry Laraway
41 N.W. Doncee Drive
Bremerton, WA 98310
206-692-5374

The BlackHawk MM/1b

Based on the AT306 board from Kreider Electronics. Features built into the motherboard include:

- 16 bit PC/AT I/O bus with five slots
- MC68306 CPU at 16.67MHz
- 512K to 16MB of RAM with 30 pin SIMMs (4 sockets)
- IDE Hard Drive Interface (2 drives)
- 360K-1.44MB Floppy Drive Interface (2 drives)
- Two 16 byte fast serial ports (up to 115K baud)
- Bi-directional parallel printer port
- Real-time clock
- PC/AT keyboard interface
- Standard PC/AT power connector
- Baby AT size - fits standard PC case
- BASIC (resembles Microsoft BASIC)
- MGR Graphical Windowing Environment with full documentation
- "Personal" OS-9/68000 Vr 3.0 (Industrial with RBF)
- Drivers for Tseng W32i and Trident 8900 VGA cards
- Drivers for Future Domain 1680 and Adaptec AAH15xx SCSI cards
- OS-9/68000 Vr 2.4 with Microware C 3.2, Assembler, MW Basic (like Basic09), MW Debug, MW Programmers Toolkit
- UUCP from Bon Billson
- Ghostscript (software PostScript interpreter)
- Many other utilities and tools

Prices start at $400!
(motherboard,
Personal OSK, & MGR only)

BlackHawk
Enterprises, Inc.
756 Gause Blvd. #29
Slidell, LA 70458
E-mail: nimitz@delphi.com

Operating System Nine

Alan DeKok

Some ins and outs of OS-9 from the NitrOS-9 programmers!

This article is number two of a series of articles on the inner workings of OS-9 Level II and NitrOS-9.

Debugging OS-9 Boot Disks

Last time we covered what is _supposed_ to happen during the OS-9 boot process after you type DOS. However, sometimes things do go wrong, and you end up with the incredibly helpful 'FAILED' message.

What can you do about it? Here are two approaches, first under OS-9 Level II, and second under NitrOS-9. Compare the steps described in this article with the last article on the OS-9 Boot process, and you'll have a pretty good idea as to what the problem is.

OS-9 Level II

You can get more information than just the 'FAILED' message, if you listen carefully to the disk drives as the system boots... Listen to the pattern of SEEKs (bzzzz) as the head of the drive moves from track to track and PAUSEs. This method works better when using a floppy disk drive,as the head seeks are easier to hear, and everything happens much slower than with a hard disk drive. The information below is listed in the format of a paragraph describing what to listen for, followed by a paragraph describing what probably went wrong if you didn't hear what you expected.

1. Upon first booting (when you type DOS after power-on), the drives will SEEK to track 34, and read the kernel track. You now get the 'OS9 BOOT' message. If you don't, then no kernel track was found, and control returns to Disk Basic (DECB). In this case, you probably don't have a bootable disk.

2. A short PAUSE, while the kernel track is verified. 'FAILED' here means you don't have a BOOT module on the kernel track.

3. A long SEEK to LSN0. This seek will probably sound much like the previous one. This means that BOOT is unable to find information about the OS9Boot file in LSN0. You must use 'OS9Gen', 'ezgen', or 'cobbler' to write out the OS9Boot file, and to update LSN0.

4. A SEEK to the OS9Boot file. If your system is working up to this point, step 3 and 4 will come *very* close together. i.e. SEEK, SEEK. Now we hear a few head clunks. On a floppy: thunk..thunk.. or on a hard drive: click..click.. while BOOT reads in the OS9Boot file. 'FAILED' here means there was an error reading the OS9Boot file, probably from a bad disk.

Usually an error here is accompanied by repeated attempts to read the problem sector, with the drive going to LSN0 between attempts. You'll hear SEEK (to LSN0), SEEK (back to the sector), and a short PAUSE while it gets an error, before the SEEK SEEK starts over. Eventually you get a 'FAILED' message, meaning the BOOT module couldn't read in one of the OS9Boot file sectors.

Remember, your OS9Boot file must be contiguous, or BOOT will load in the wrong information, and OS-9 will not run (Commands Ref 6-70)! The reason is that the information in LSN0 points to the OS9Boot *file*, and not its OS-9 RBF-controlled File Descriptor. In effect, LSN0 contains information for only the first segment of the OS9Boot file, so that is the only segment BOOT can use (Tech Ref 5-3). It may be possible to allow LSN0 to point to the OS9Boot file descriptor, which contains the segment information, but that would force ALL versions of the BOOT module to be rewritten, along with a cobbler, OS9Gen, ezgen, and similar programs. It would also require new information in LSN0 to point to the OS9Boot file descriptor.

The current system of finding the OS9Boot file does work, and many versions of BOOT do not have enough room to add new routines, so there's not much likelihood of things changing.

5. Next, a LONG PAUSE while the modules in the OS9Boot file are verified. 'FAILED' here means you don't have one of the following modules in your OS9Boot file (in order of probability):

'D0' or 'DD'... (device descriptor pointed to by the 'INIT' module.)

'CC3Disk' or 'BBHDisk'... (the device driver for /DD.)

'Init'

'OS9p2'

Now it gets complicated... Steps 6, 7, and 8 are usually close enough together that all you hear is a bunch of SEEKs in no apparent pattern. If the system crashes ('FAILED'), however, you can try to cou the SEEKs to get an idea of what's wrong.

6. A SEEK, SEEK while OS9p2 does a CHX, and opens /TERM. More SEEKs and a short PAUSE while GrfDrv is loaded and verified. 'FAILED' here can mean different things..

a - there was an error doing the CHX, or reading the root directory.

b - the device descriptor 'Term' is not in your OS9Boot file.

c - the driver for /Term (usually CC3IO) is not in your OS9Boot file.

d - If Term is controlled by WindInt, GrfDrv may not be in /DD/CMDS, or

e - GrfDrv may not have it's execution attribute set (attr grfdrv e pe)

7. SEEK SEEK while OS9p2 loads CC3Go (assuming CC3Go isn't in your OS9Boot file). CC3Go then prints out the sign-on banner, and you see the OS-9 copyright message on your OS-9 '/Term' window. 'FAILED' here means CC3Go does not exist, or it's attributes are wrong (attr

CC3Go e pe).

8. CC3Go does a lot of things, CH CHD, another CHX, and then loads SHELL. 'FAILED' here means many things..

a - there was no CMDS directory. (Not likely, since GrfDrv was found, unless you are booting to a VDG type window.)

b - Shell was not in the CMDS directory.

c - Shell did not have it's execute permissions set (attr shell e pe)

If you see a flash of the sign-on message, and then 'FAILED', it probably means that something's gone wrong with finding Shell. Check the three items above to verify that Shell exists, and everything should be OK.

9. Now your 'Startup' file is executed. If it gets this far, your OS-9 system is up and running. Even if 'Startup' doesn't exist, you will at least get the shell prompt.

That covers most of the problems you are likely to encounter. Next time you get the dreaded 'FAILED' message, you'll be prepared.

NitrOS-9

NitrOS-9 v1.20 and following make debugging boot disks much easier. As a

boot progresses, information about each step is printed to the boot screen. In the event things crash seriously, you can even access the information from DECB!

When booting NitrOS-9, you will see a series of characters printed out on the boot screen. The NitrOS-9 manual describes these characters and their meaning, so we won't do much more than summarize that information here.

Every module that is verified as part of the OS-9 boot process has its module name printed out on the boot screen. If you follow the boot steps described above and see the 'FAILED' message, you can simply look at the boot screen above the FAILED message to find out which module you are missing.

In addition to the module names, there are one-character status reports printed out. These characters tell you everything from which module (OS9p1, OS9p2, or CC3Go) is currently controlling the boot process, to which system calls CC3Go is performing as it tries to boot.

Depending on what is wrong with the boot, you may see an asterix '*' as the second last character printed out. The character following the asterix is the OS-9 error code that was returned during a critical system call, which caused the system to abort the boot.

Once the system has booted completely, the information on the boot screen will still be in memory. If you have the 'ded' program, the 'md' device descriptor and the 'rammer' driver, you can 'ded /md@'. Go to sector $760, and you will see the boot debugging characters, starting at offset $08.

If your system REALLY crashes, go back to DECB (CTL-ALT-Reset), and type the following line:

```
FOR A = &H6008 TO &H6100:PRINT CHR$(PEEK(A));:NEXT
```

All of the boot debugging characters will be printed out for your inspection. If the characters appear to be random garbage, you have probably been rewriting system code, and have introduced bugs to the system! Correct your code, and try again.

As you can see, debugging boot disks is a LOT easier this way. A small amount of memory is lost from the system by adding the debugging code, but we believe that the trade-off is worth it. The memory used up again is only a tiny portion of that saved by running NitrOS-9 in the first place.

The Zen of CoCo Programming

continued from page 10

The 'shell' has commands similar to DECB (RS-DOS) like "cls" for clearing the screen, 'dir' for taking directories, "load" for loading executables and "exec" for running the executables. It also contains commands such as "reset" to reset the system in case something goes wrong. Unlike DECB, commands do not have to be in uppercase. They can be upper, lower or a mix. They also do not require quotes around filenames in order to work. The shell allows input from either the CoCo keyboard, the serial port or both.

So now that you know what it has in it, how does it work? Like any other type of OS, when it first loads (either from ROM or disk), it sets up both the user and system stacks, initializes all hardware that it supports (it is available) and executes the command line shell.

The CDS takes it a few steps further. Before anything is initialized, it sets up the screen display so that diagnostic information can be seen during the startup process. Immediately after setting up the stack, it goes out and does a hardware check to ensure that certain hardware devices are present. It searches for hard drive controllers, RS-232 Paks (in 2 different address ranged), the Multi-Pak Interface and a few other minor things.

Once all hardware devices are identified, they are initialized and reset to default settings (such as baud rate for the serial ports and track 0 for floppy drives). After all of the hardware is initialized, the CDS accesses the serial ports (RS-232 Pak) if they are available and searches for the host machine. If a host is found, it requests some information and sets a few internal parameters. If a host is not found it continues with setting up the system. Once all of the set up is done, it goes into the command shell and waits for user input.

In its current form, the CDS is less than 8k in size including data buffers and can be burned into ROM. Unfortunately, it is far from being finished. There are several aspects of the CDS that have strayed from its original designed. All of those items are going to be changed before it is released for general use. My current plans for the CDS entail changing all commands to loadable modules, adding a module directory structure similar to that of OS-9's, an extensive device manager and adding system calls that are accessed with SWI instructions instead of the current vectored method.

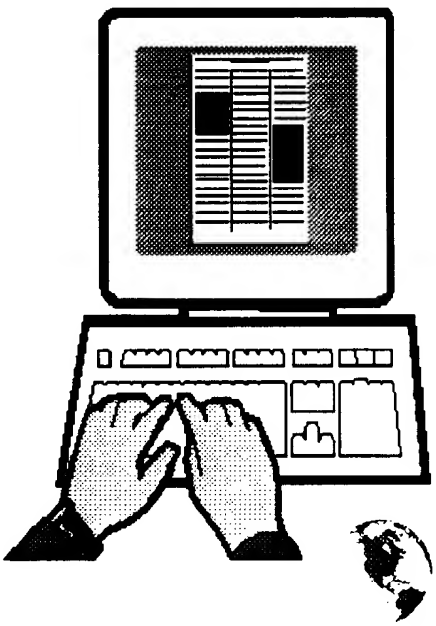
In future articles we will be touching base on programming interrupt routines, graphics hardware, high-speed graphics routines, sound and code optimization. We will also delve into writing those simple but needed routines like converting binary values to ASCII numbers, screen output, file I/O and many others. In each article I will also keep you updated on the progress of the CDS and once it is complete we will spend a couple of articles discussing how to write programs that use it and where it can be obtained.

If you have access to the interenet and have and comments, questions or suggestions, you can conact me via e-mail at the following address:

chets@corenet.net

The CoCo lives at:

http://www2.corenet.net/medialink/coco/coco_main.html



Modifying Process Descriptors

Mark Heilpern

No trick, just do it in system state!

I want to write a OS-9 program similar to Norton Change Directory, an MS-DOS utility. But the Process Descriptor is not accessed on user-state. In order to write such a program, I need to know how to:

(1) modity a Process Descriptor.

(2) get the Parent Process ID.

(3) get the Process Descriptor's pointer from the Process ID.

I also need to know the detailed contents of Process Descriptor.

Process Descriptos must be modified in system state. Since changing a directory is not likely to take a long period of time, other programs shouldn't be affected.

To get the parents process descriptor address you need to know the id of your parent (which is in your process descriptor, field _pid). You also need the table of process descriptors which the system global d_prcdbt points to. Then just index into that table to find the parent.

See the header file <procid.h> (or, process.h if this is for OS-9000) for the 'procid' structure. In your 'ncd' program, I assume you'll be changing your own directory, then copying the directory change to the process descriptor of the parent process. The fields to copy are not exactly obvious; you'll want to copy the _dio array. The first half of this array contains information for the 'data' directory, the second for the 'execution' directory.

Here is a program I wrote quite a while back to implement a stand-alone 'cd' utility. It gets around the memory protection problem by calling _os_permit() to get access. For this to work the program must execute as super-user (group 0). If you will not always run this as super-user you must modify the code somewhat (make the module owned by group 0 and toss in a _os_setuid() to change yourself to group 0 early in the program). If you have no MMU or are not running the SSM extension there is no memory protection.

THIS CODE IS NOT GUARANTEE'D TO WORK, BUT IT WORKED THE LAST TIME I COMPILED IT!

```
/*** This is the first of 2 files ***/
#include <types.h>
#include <stdio.h>
#include <process.h>
#include <errno.h>
#include <cglob.h>
#include <modes.h>
error_code find_proc_desc(process_id,
procid **);
main(u_int32 argc, char **argv)
{
   char *pathname = argv[1];
   u_int32 mode = S_IREAD;
/* change data directory */
   procid *me, *dad;
 /* if no directory was specified, check
for default */
        if (argc==1) pathname =
(char*)getenv("HOME");
           if (pathname==NULL)
exit(E_BPNAM);
/* check for execution directory change
request */
   if (argc>2)
   {
         if (!strcmp(argv[1],"-x"))
pathname = argv[2];
      mode = S_IEXEC;
/* change execution directory */
   }
/* do the directory change */
   errno = _os_chdir(pathname,mode);
   if (errno) exit(errno);
 /* find my process descriptor */
    errno = find_proc_desc(_procid,
&me);
   if (errno) exit(errno);
/*find my parent's process descriptor */
    errno = find_proc_desc(me->_pid,
&dad);
   if (errno) exit(errno);
/*copy over the directory information */
    errno = _os_cpymem(_procid,&me-
>_dio,&dad->_dio,DEFIOSIZE);
   if (errno) exit(errno);
/* and exit */
   exit(0);
}

/*** the next file is for the
find_proc_desc() function ***/
#include <process.h>
#include <sysglob.h>
 /* to get system globals */
#include <stddef.h>
/* for 'offsetof() macro */
#include <modes.h>
 /* for permit access modes */
#include <errno.h>
extern process_id _procid;
/* my id */
/*
** Usage:
**     process_id proc_id;
**     procid *proc_desc;
**          errno = find_proc_desc
(proc_id,&proc_desc);
*/
error_code find_proc_desc(process_id
proc_id,
   procid **proc_desc)
{
   u_int32 *ptab;
   u_int32 size;

   /* first, find the system's process */
   /* database table */
        (void)_os_getsys((offsetof
(sysglobs,d_prcdbt)),sizeof(u_int32*),
(glob_buff*)&ptab);

/* get access to this memory region *'
/* number of bytes we need access to
/* (size) is the process id of interest, */
/* times size of each pointer entry (4) */
/* plus the size of one entry (4) */
   size = (proc_id+1)*4;
                          errno =
_os_permit(ptab,size,S_IREAD,_procid);
   if (errno) return(errno);
/* got the table. lets index into it */
  *proc_desc = (procid*)ptab[proc_id];
/* finally, get access to that memory */
 (void)_os_permit(*proc_desc,sizeof(procid),
     S_IREAD|S_IWRITE,_procid);
/* note, don't need error checking on */
/*the last _os_permit(),since the call */
/*should only fail if not running */
/*as super user. since the first permit */
/*call worked, we must be a SU */

   return(0);
}
```

Questions? I can be reached via e-mail at: **heilpern@microware.com.** *If you don't have e-mail access, feel free to write the editor in reference to this article.*

RELEASE-PRESS RELEASE-PRESS RELEASE-PRESS

GESPAC ANNOUNCES G-WINDOWS 3.0

Mesa, March 22, 1996 — GESPAC has released version 3.0 of G-WINDOWS. G-WINDOWS is a windowing software package that is destined to add a Graphical User Interface to real-time embedded systems. With G-WINDOWS an embedded controller that is using a real-time kernel could also be equipped with a graphics controller for the user interface. A single CPU is now capable of controlling the display at the same time that it is managing other real-time processes.

G-WINDOWS is now available for OS-9 68K and OS-9000 PC platforms. It is currently being ported to VxWorks and other real-time OS's will follow. G-WINDOWS requires as little as 250K of memory to operate, compared to 10+ megabytes for MS Windows or X-Windows, and can run from ROM or disk. G-WINDOWS is ideally suited to run on target embedded systems based on the VME, G-64, PC, or fully custom architecture. G-WINDOWS is also well suited for manufacturers of set-top boxes, PDAs and Multimedia systems. **Product information and demonstration screens can be downloaded free from GESPAC's Internet site at http://www.gespac.com.**

G-WINDOWS is composed of a file manager/driver that is intimately linked to the OS kernel and which supports a broad selection of graphics primitives (lines, circles, rectangles, polylines, etc.), multiple fonts, including Japanese, Chinese and Korean characters, and window management primitives (frames, scroll bars, event management, etc.). A C library is available for the programmer who wants to write programs using low level calls to the file manager/driver in order to draw a screen and interact with it.

G-WINDOWS also supports the G-VIEW editor which allows the user to easily edit sophisticated interface screens, complete with buttons, needle gages, bar graphs, X-Y plots, moving images, and some 20 other types of I/O gadgets. The I/O gadgets can then be linked to te user's application program written in C with as little as a single line of C code for each gadget.

G-WINDOWS is supported with hardware provided by GESPAC and other companies. For the VME bus, GESPAC offers two graphics controllers and two Industry Pack compatible VGA modules for CRT or LCD. For PC systems, G-Windows supports the most popular VGA controller cards as well as an accelerated 64-bit PCI VGA card.

G-WINDOWS was first released in 1990 and is broadly used in machine control, medical instrumentation, process control and many other applications requiring real-time control performance and a graphical user interface that operates concurently on the same machine. Version 3.0 bring several improvements to the software, including new I/O gadgets, support for Japanese, Chinese and Korean languages, and better real-time performance.

Founded in 1979, GESPAC is a leading manufacturer of industrial microcomputer board level products with a catalog of over 500 product references and sales in Europe, North America and the Pacific Rim. GESPAC is part of the COFIDUR Group, a 1000 person, $200 million strong conglomerate of companies specialized in electronics design, PCB manufacturing and assembly, and system integra

Start making plans to attend now!

THE SIXTH ANNUAL "LAST" CHICAGO COCOFEST

TO BE HELD IN APRIL, 1997, AT THE EGLIN HOLIDAY INN

Will this REALLY be the last CoCoFest? You DON'T WANT TO MISS IT!!!

For reservations call the Eglin Holiday Inn directly at 708-695-5000. The special 'fest rate is only $57 per night. A limited number of rooms will be held for 'fest participants, so make your reservations as soon as possible. Tickets are $5 each for Glenside members, $10 for all others. Of course this would be a good time to join Glenside and get the club newsletter! Membership is only $15 per year. Send ticket and/or membership fees (personal checks or money orders) to:

George Schneweiss, Treasurer
Glenside Color Computer Club
RR#@ Box 67
Forrest, IL 61741-9629

For additional info and vendor inquiries, contact Eddir Kuns (708-820-3943; e-mail eddiekuns@delphi.com) or Tony Podraza (708-428-3576 or BBS 708-428-0436, 8-N-1, 2400 baud).

The Eglin Holiday Inn is a **Holidome Family Rec Center** with indoor heated pool, games, and excercise room. Enjoy!!!

You are cordially invited to the first Pennsylvania

COLOR COMPUTER SHOW & SALE

WHEN? August 2 & 3, 1997 (Sat. 10am-5pm; Sun. 10am-3:30pm)
WHERE? EMBERS INN 1700 Harrisburg Pike, Carlisle, PA

ADMISSION:
$5.00 per person per day *OR* $7.00 for both days (paid in advance). Children accompanied by a responsible adult are FREE!

HOW TO GET THERE:
Exit 16 off of the PA Turnpike I-76, turn to Harrisburg, go 1.3 miles, it's on the right *OR* I-81 Exit 17, turn left, go 1/10th mile, on the right!)

Overnight room rate:
$60 - Be sure to ask for the "FEST" rate!
Call 1-717-243-1717 OR 1-800-692-7315 OR Fax (717) 243-6648 for reservations! There is a limited supply of rooms blocked out for the show. Reserve your room early - - these rooms will be released for regular reservations on July 1, 1997 and will NOT be available at the show rate after that date!

For further information, General or Exhibitor, contact:

Ron Bull
115 Ann Street
Duncannon, PA 17020

Phone: 717-834-4314
E-mail: ronbull@aol.com *OR* ron.bull@paonline.com

Chris Dekker's CoCoTop

Frank Swygert

An easy to use alternative to Tandy's Multi-Vue

Some time ago, Chris Dekker released a very interesting desktop file management program called "CoCoTop". As the name implies, it is for the CoCo. It works with the OS-9 Level II operating system. The whole idea behind the program is to take a lot of often used functions and put them into one easy to use interface along with some of your own often used programs.

Before I go any further, I must state that I did not write most of this review. It is pretty much paraphrased from Chris' excellent documentation, mainly from his introduction.

CoCoTop uses a simple text based interface with terms such as copy, move, erase, etc., to identify the various functions built into the program. A mouse or joystick is used to activate most of these functions, although some (such as entering paths and filenames) require keyboard interaction.

Accessing programs buried deep in disk directories is also made easier. CoCoTop allows a user to simply click on the drive and directory, then choose the program to run. The interface tracks the path, even opening a new window for the selected program to run in, and most of the time closing it when the program is ended.

Are you tired of seeing all the meaningless error messages when you try to run a program? CoCoTop traps all errors. Chris has replaced the simple error numbers OS-9 normally provides with more meaningful messages, telling you what most likely went wrong.

Unlike many OS-9 programs, this one was written with the novice user in mind. While one will still need to create a boot disk and install all the extra utilities, CoCoTop is pretty easy to install. Chris wrote an installation script that is pretty easy to follow. Simply follow the instructions on screen after running the *CCTsetup* program.

The documentation for CoCoTop is some of the best I've seen for any utility software. All instructions are pretty easy for beginning to intermediate users to follow. Chris does, however, expect one to know a little about OS-9. If you have just started to use OS-9 don't get to frustrated-- just keep your OS-9 manual handy to help you through.

The main functions included in CoCoTop are listed below. Each has several subfunctions:

DISK: eleven disk related functions (select drive, disk info, disk map [a semi-graphic showing of disk use], reset markers [directory markers used by CoCoTop], clear disk [a reversible quick format], unformat [works only on disks formatted with clear disk], format [semi-graphic interface for OS-9's forma]), ramdrive [a ramdrive setup utility], backup, check disk [runs dcheck], and rename disk).

TREE: five functions to ease travel through disk directories.

DIRECTORY: nine directory functions (update screens [reread directory], make new dir, delete dir, clear dir [deletes files but not directory], copy contents, move contents, sort contents, rename dir, move dir, change attr).

FILES: fourteen file handling functions (information, copy, move, delete, undelete [only works with files dleted from CoCoTop], erase, rename, compare, view contnets, print contents, edit contents, append, split, change attr).

FIND: a file finder (search) function.

SYSTEM: eleven system control functions (pause on/off, system info, printer, disk drive, memory [shows how much you have, how much is free, and a semi-graphic map], module list, process info, date/time, alarm, reconfigure [forces CoCoTop to reconfigure itself, useful if modules installed after CCT is running].

ACCESSORIES: Nine extra functions that just didn't fit anywhere (shell [command line], run program, load modul' unlink module, setup window, remove wi. dow, screensaver [blank screen with randomly moving "press mouse button" message], calculator, notepad.

I mention "semi-graphics" several times. This isn't the old semi-grpahics mode of the CoCo 2, but graphics symbols from a highly modified version of Floyd Resler's public domain gfx5 module. This is included with CoCoTop so one doesn't have to go looking for it anywhere.

Chris Dekker did an excellent job of programming CoCoTop. It is very functional and easy to use. It is sold with or without his TOOLS3 package: $19.95 without and $34.95 with (TOOLS3 is $29.95 alone). Several functions (append, erase, password, alarm, dircopy, dupdisk, split, and backup) are required from TOOLS3. CoCoTop will function, however, without them.

For more information or to order, check Chris's ad below.

EDTASM6309 v2.02 - $35.00
Patches Tandy's Disk EDTASM to support Hitachi 6309 codes! Supports all CoCo models, including stock 6809 models. CoCo 3 version uses 80 column screen, runs at 2MHz. YOU MUST HAVE A COPY OF DISK EDTASM. This is a PATCH ONLY! It will not work with "disk patched" cartridge EDTASM

CC3FAX - $35.00
Receive and print weather fascimile maps from shortwave! The US weather service sends them all the time! Requires 512K CoCo3 and shortwave receiver. Instructions for simple cable included.

HRSDOS - $25.00
Move programs and data between DECB and OS-9 disks! Supports RGB-DOS - move files easily between DECB and OS-9 partitions! No modifications to OS-9 modules required.

DECB SmartWatch Drivers - $20.00
Access your SmartWatch from DECB! Adds function to BASIC (DATE$) for accessing date and time. *Only $15.00 with any other purchase!*

RGBOOST - $15.00
Make the most of your Hitachi 6309 by using it with DECB! Uses 6309 functions for up to 15% gain in speed. Compatible with all programs tested to date! *Only $10 with any other purchase!*

Robert Gault
832 N. Renaud
Grosse Pointe Woods, MI 48236
313-881-0335
Please add $4 S&H per order

Northern Xposure

NOTICE

Due to many complaints about delayed or unprocessed shipments and virtually no contact, we are no longer printing advertising for Northern Exposure. Should this situation change, I will reinstate advertising for this company. For those who have ordered because of these ads, I can only apologize. I have contacted Northern Exposure about the order delays and have made arrangements to process unfilled orders for NitrOS-9, TuneUp, Thexder, and Shanghai. If you have outstanding orders, send me a copy of the cancelled check and a $5 handling fee and I'll forward the software. Again, my sincere apologies to readers.

Frank Swygert, Editor

for all your CoCo hardware needs, connect with

CoNect

449 South 90th Street
Milwaukee, WI 53214
(pulland@omnifest.uwm.edu)

That thing that Tandy calls a serial port on the CoCo has always been a problem. It was designed with minimal cost in mind, and never upgraded. Even Tandy tried to fix it with their RS-232 Pak, but even it was only half done! Our Fast 232 port uses a 16 byte buffer at alleviate missed characters at any speed and also has ALL RS-232 lines implemented. It is easy to set up with jumpers for different addresses. A daughterboard can be purchased to easily add a second fast serial port! And all this in a cartridge the size of a ROM Pak! 6809 and 6309 OS-9 drivers included. Completely supports up to 57,600 bps, limited support for 115,000 bps.

Fast 232 - $79.95
Daughter Board - $45.00

Check with us for complete disk drive systems, misc. hardware items, hardware repairs, and hard to find new and used CoCo software!

ADVERTISER'S INDEX

BlackHawk Enterprises	*11*
C. Dekker	*16*
Chicago CoCoFest	*15*
CoCo Show & Sale	*15*
CoNect	*17*
FARNA Systems	*7, 10*
Robert Gault	*17*
Hawksoft	*11*
Northern Exposure	*17*
Small Grafx	*11*
Wittman Computer Products	*BC*

NEWS FLASH!!!

FARNA Systems is now the distributor for:
NitrOS-9
TuneUp
Thexder OS-9
Shanghai OS-9
See the FARNA Systems ad on page 7 for prices and details!

All About ROM Packs

Marty Goodman

Transferring ROM Packs to disk and understanding large ROM Packs.

Some ROM paks can easily be transferred to diskette. Tennis, Backgammon, Chess, Checkers, Polaris, Project Nebula, Poltergeist, Skiing, etc., are all TRIVIAL to put on diskette, for NO attempt WHAT SO EVER was made to prevent running them out of the top 32K of RAM.

Megabug by Steve Bjork was, if I recall correctly, the FIRST EVER "protected" ROM pak. It contains code that SPECIFICALLY was designed to thwart having the ROM pak run in RAM. Typically such code (in the case of CoCo 1 and 2 ROM paks) consists of code that WRITES OVER part of the ROM pak code. This of course has no effect if the code is running IN a physical ROM (since such writes will have no effect), but "stabs" parts of the program if the program is running in RAM.

Added levels of sophistication for such protection includes making the act of doing the writes relatively hard to spot, by using non-obvious looking code and/or burying the stabbing code in a large perfectly normal, functional, and legitimate part of the game's code. Yet another level of such ROM pak protection includes having code totally elsewhere (and similarly obscure and hidden in normal code) that CHECKS for the integrity of the stabbing code, and halts or crashes the game IF this other code detects you've tried to disable the primary stabbing code.

Megabug was "cracked" shortly after it was released by more than one CoCo hacker. The cracking consisted of patient examination of the game code until the self-destruct code was found. Megabug, if I recall correctly, was relatively easy to crack... the protection it had was not that sophisticated.

Canyon Climber was first cracked by Jeff Francis. I believe I was with him as he did the analysis (disassembly) of the game's code and found the one or two or three points of self-stabbing code. It took him about an hour to finish the crack.

Steve Bjork later incorporated more nasty protection into his Stellar Life Line. It took a very accomplished CoCo assembly language programmer and disassembler and game cracker (Mike Ward) quite a while (days) to root out the several self-stabbing and/or checking routines in that code and produce a version that ran from diskette. Part of what made it hard was that Steve put in several self-stabbing code pieces at different LEVELS of the game... you had to play the game with some skill to encounter the next bit of trick code.

Steve Bjork's Arkenoid and Rampage for the Coco 3 present yet another kind of problem: They use ROMs that are bigger than the ROM area of the CoCo, and have hardware bank select circuitry inside the ROM pak to acess all of the ROM. Both of those were successfully cracked by a chap on Delphi, who used the extra memory in a 512K CoCo to simulate the extra banks in the bank switched ROM. He had to find every place the bank switcher was used, and substitute his code that used the MMU instead of the bank switcher.

Robocop used the biggest ROM of all... a 128K by 8 ROM, divided into eight 16K segments, if I recall correctly, using a hardware bank switcher. In theory this should be crack-able using the same approach as that used to crack Arkenoid and Rampage, tho in practice I've yet to hear of ANYONE who's successfully put Robocop on diskette to run on a 512K CoCo 3. It is the ONLY such "never cracked" CoCo 3 ROM pak that I know of.

Of course, nothing stopped the authors of bank switched ROM paks from incorporating OTHER protection (self stabbing code, checks on the self stabbing code, etc.). I have no idea if Robocop requires merely emulation of the bank switcher OR whether there are other hurdles the author put into the code to prevent running it off diskette.

Activison Bank Switching

Activision packs use a 74LS10 triple 3-input NAND gate and a 74LS175 edge triggered quad D flip flot to switch up to 16 banks. In practice, it appears that in Predator only four banks are used (a 64K by 8 ROM is divided into 4 16K banks) and in Robocop eight banks are used (1 Megabit 128K by 8 ROM is switched into 8 16K banks. Switching is accomplished by writing to the first four bits of $FF40. This sets the outputs of the four flip flops in the LS175. Those outputs control the high order address lines of the ROM. At power up or hardware reset, the LS175 is set to all zeros on its output, due to its master reset being wired to the CoCo master reset. Thus, at power up, the first 16K bank is automatically guaranteed to be selected.

The ROM is wired pretty much as a normal 16K CoCo ROM is wired, with D0 thru D7 of the CoCo system bus going to D0 thru D7 on the ROM, and A0 thru A12 of the CoCo system bus going to A0 thru A12 of the ROM, in standard 27 series JDEC pin out on the ROM. Ground, Vcc, and Chip enable (pin 20) of the ROM are also wired as usual to ground, Vcc, and *CTS of the CoCo system bus. Pins 27, 1, and 22 of the ROM, however, (normally A14, A15 , and either Output Enable on a 64K ROM or A16 on a 128K ROM) are controlled by the LS175 flip flop, as indicated in the diagram.

Note that writes to ANY address in the range of $FF40 thru $FF5F will set the bank select latch, for the *SCS of the CoCo is NOT decoded any further at all. Note, too, that in these two Activison ROM packs, although they COULD have used a 32K bank size (because the Coco 3 can address directy up to 32K of ROM in its port) they actually elected to NOT hook up A14 of the CoCo system bus to the ROM, and so instead choose a 16K bank size. This, perhaps, to provide for future support of cartridges that would run on both the Coco 3 and the CoCo 2, for the latter cannot address more than 16K in its ROM pak port.

What does all this mean?

Activision has provided for BANK SWITCHING of the ROM. Writing data in to address $FF40 will select differing banks of the ROM, for it sets the status of high order address lines of the ROM. Moreover, Activison elected to select the ROM in banks of 16K.

The Predator ROM uses 4 banks of 16K. That is, it has a total of 64K bytes of data. The RoboCop ROM can be selected into 8 banks of 16K, or 128K ROM pack!

To examine all the data in the ROM, you merely need to cover pin 7 or 8 (to defeat the auto execute), then use your favorite ROM dump technique to dump of the first 16K of the ROM. THEN, write 1 into $FF40, and dump the next 16K the same way, for it will reside at $C000 thru

$FFFF. Go along writing 2, and 3 into $FF40 to get all of the Predator ROM. You will have to continue and write 5 thru 7 into the RoboCop ROM to get all eight 16K banks.

Making these ROMpaks run off disk will involve loading all of their data into a 512K CoCo, then replacing the bank select routines that write to $FF40 with your own subroutine that uses the MMU to shuffle the data around, in addition to the usual breaking of "protection" and getting around the turning on and off of the ROM at $FFDE / $FFDF.

Marty Goodman

Editors notes: Patches for many ROM paks can be found on Delphi and various CoCo BBS's and web sites. Instructions for transferring ROM paks to disk can also be found there. This issue's "microdisk" will also have some patches on it. If by chance anyone HAS cracked Robocop, please inform this magazine so we can pass the word as to how it was done.

The following program will transfer ROM Packs that don't need cracking. Although it should be used with a Multi-Pak, one can transfer programs to disk without the MPI on a CoCo3. Do the following:

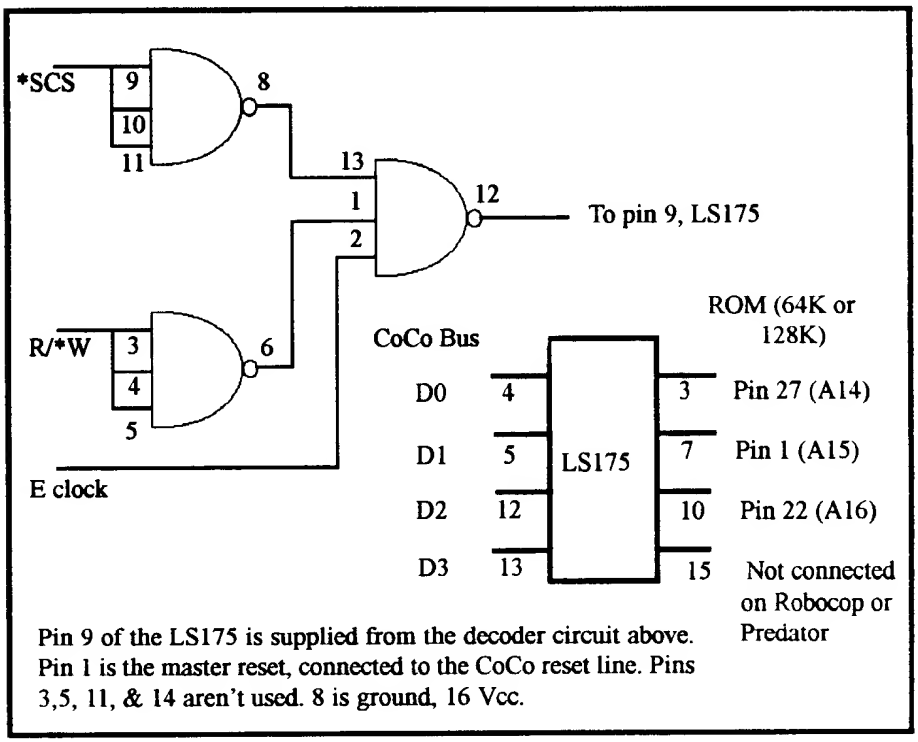
1) CLEAR200, &H3FFF
2) POKE &HFF23,&H36
3) Carefully insert ROM Pack

WARNING You must be VERY CAREFUL when inserting and removing a ROM Pack with the computer on! If it is moved at an angle, it could cause a short and blow the 6809. Insert carefully and STRAIGHT in and out and you should have no problem.

4) CSAVEM"gamename", &H4000, &H7FFF, &HB44A
5) Power computer off and insert disk controller
6) CLEAR 200, &H3FFF
7) CLOADM"gamename"
8) SAVEM"gamename.bin", &H4000,&H7FFF, &HB44A

Alternately (and this will work for CoCo 1 and 2 systems as well), place a piece of scotch tape over pin 8 of the ROM Pack (and only pin 8). With power off, insert the ROM Pack into the CoCo and turn the computer on. The tape over pin 8 will prevent the ROM Pack from auto execution. When you get the EXTENDED COLOR BASIC message, type POKE &HFFDE,0 and press enter. Now save to tape using CSAVEM "gamename", &HC000, &HFEFF, &HA027. Turn the machine off, remove the ROM Pack, and reinstall your disk controller. Power up the CoCo and load the tape just made by typing CLOADM "gameneame", &H6000. Now save to disk by typing SAVEM "gamename", &H2000, &H5EFF, &H2000. To play the game simply LOADM"gamename" and EXEC.

```
0 REM THIS PROGRAM WILL COPY 16K OR
32K ROM PAKS TO DISK AND MUST BE
USED WITH A MULTIPAK. WHEN ENTERING
FILENAMES, DON'T USE AN EXTENTION.
THE PROGRAM WILL ASSIGN EXTENTIONS
AS FOLLOWS:
1 REM 16K ROMS ARE ONLY 1 FILE LONG
AND THE EXTENTION IS /ROM. 32K ROMS
ARE IN TWO FILES. THE LOWER 16K ($8000-
$BFFF) HAS THE EXTENTION OF /LWR. THE
UPPER 16K ($C000-$FEFF) HAS THE
EXTENTION OF /UPR.
2 REM YOU NEED ONLY ENTER 1 NAME
FOR ANY FILE (16K OR 32K).
3 REM BE SURE THAT YOUR DISK CON-
TROLLER IS IN SLOT FOUR ON YOUR MULTI-
PAK INTERFACE.
4 REM ROM PACKS CAN BE IN ANY OF THE
OTHER SLOTS (1-3). BE SURE TO INSERT
THE ROM INTO THE MULTIPAK BEFORE
POWERING UP THE HARDWARE. DO NOT
MOVE THE SELECTOR SWITCH ON THE
MULTIPAK INTERFACE.
5 REM THIS PROGRAM DOES NOT MODIFY
ROM CODE, IT JUST MOVES THE CODE
FROM ROM TO A DISK FILE
6 REM I HOPE THIS PROGRAM WILL BE
USEFULL FOR YOU IN SIMPLIFYING THE
ROM COPYING PROCESS. HAPPY COPY-
ING!!!
7 REM
8 GOTO31
9 DEFUSR0=&H2000
10 CLS:POKE&HFF23,&H36
11 PRINTTAB(3)"COCO ROM COPY UTILITY"
12 PRINTTAB(3)"* BY DAN M. AUSILI & *":
PRINTTAB(3)"* RON ZBOROWSKI (C) 1989
*":PRINT
13 INPUT"SLOT NUMBER";SL$
14 LETSL=VAL(SL$):IFSL<1 OR SL>4 THEN 11
15 LETSL=SL*16-16+3
16 X=USR0(SL)
17 IFX=16 THEN PRINT"16K ROM COPIED":
GOTO18 ELSE PRINT"32K ROM COPIED"
18 PRINT:INPUT"ENTER FILENAME";N$
19 PRINT:PRINT"INSERT TARGET DISK IN
DRIVE 0"
20 PRINT"PRESS 'ENTER' WHEN READY"
21 I$=INKEY$:IFI$<>CHR$(13)THEN21
22 IFX=16 THEN N1$=N$+"/ROM":GOTO27
23 N1$=N$+"/LWR"
24 N2$=N$+"/UPR"
25 POKE&HFFA2,&H32:POKE&HFFA3,&H33
26 SAVEM N2$,&H4000,&H7EFF,&HB44A
27 POKE&HFFA2,&H30:POKE&HFFA3,&H31
28 SAVEM N1$,&H4000,&H7FFF,&HB44A
29 POKE&HFFA2,&H3A:POKE&HFFA3,&H3B
30 PRINT:PRINT"ROM COPY
COMPLETED":END
31 CLEAR200,&H3FFF:VERIFYON
32 FORX=&H2000 TO &H208E
33 READZ:POKEX,Z:NEXT
34 GOTO9
35 DATA &H34,&H51,&H1A,&H50,&HBD,&HB3,
&HED,&HF7,&HFF,&H7F,&H86,&HC7,&HB7,&HFF,
&H90,&H7F,&HFF,&HDE,&H8E,&H80,&H00,&H8D,
&H6A,&H34,&H02,&H8E,&HC0,&H00,&H8D,&H63
,&HA1,&HE0,&H26,&H0A,&HCC,&H3E,&H3F,&HFD,
&HFF,&HA4
36 DATA &H86,&H10,&H20,&H02,&H86,&H20,
&HA7,&H8C,&H5E,&HCC,&H30,&H31,&HFD,
&HFF,&HA2,&H8E,&H80,&H00,&HCE,&H40,&H00,
&HEC,&H81,&HED,&HC1,&H8C,&HC0,&H00,
&H25,&HF7,&HA6,&H8C,&H46,&H81,&H10
37 DATA &H27,&H12,&HCC,&H32,&H33,&HFD
,&HFF,&HA2,&HCE,&H40,&H00,&HEC,&H81,&HED,
&HC1,&H8C,&HFF,&H00,&H25,&HF7,&HCC,&H3C,
&H3D,&HFD,&HFF,&HA4,&HCC,&H3A,&H3B,&HFD,
&HFF,&HA2,&H86,&HCC,&HB7,&HFF,&H90,&H86,
&H33,&HB7,&HFF,&H7F
38 DATA &H7F,&HFF,&HDF,&H4F,&HE6,&H8C,
&H13,&HBD,&HB4,&HF4,&H35,&HD1,&H5F,&H6F,
&HE2,&HA6,&H80,&HAB,&HE4,&HA7,&HE4,&H5A,
&H26,&HF7,&H35,&H82
39 GOTO5
```



Pin 9 of the LS175 is supplied from the decoder circuit above. Pin 1 is the master reset, connected to the CoCo reset line. Pins 3,5, 11, & 14 aren't used. 8 is ground, 16 Vcc.

Wittman Computer Products

Now all new ways to contact us!

Our Web Page:

http://www.frontiernet.net/~wittman/wcp.html

New e-mail address:

wittman@frontiernet.net

24 hour Fax line:

(716) 494 - 2875

Voice mail 8 am to 10 pm EST:

(716) 494 - 1506

Or U.S. Postal Service:

Wittman Computer Products
Wm. L. Wittman
39 South Lake Avenue
Bergen, NY 14416

New Catalog available: October 1, 1996

All NEW and updated.